UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/755,405	01/05/2001	Brian Gerard Goodman	TUC920000051US1	5242

46917 7590 11/30/2006

KONRAD RAYNES & VICTOR, LLP.
ATTN: IBM37
315 SOUTH BEVERLY DRIVE, SUITE 210
BEVERLY HILLS, CA 90212

EXAMINER	
ZHEN, LI B	

ART UNIT	PAPER NUMBER
2194	

DATE MAILED: 11/30/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	**Application No.** 09/755,405	**Applicant(s)** GOODMAN ET AL.	
	Examiner Li B. Zhen	**Art Unit** 2194	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *11 September 2006*.

2a)☒ This action is **FINAL**. 2b)□ This action is non-final.

3)□ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-48* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)□ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-8,10-22,24-36 and 38-48* is/are rejected.

7)☒ Claim(s) *9,23 and 37* is/are objected to.

8)□ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)□ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *05 January 2001* is/are: a)☒ accepted or b)□ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)□ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)□ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)□ All b)□ Some * c)□ None of:

1.□ Certified copies of the priority documents have been received.

2.□ Certified copies of the priority documents have been received in Application No. _____.

3.□ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) □ Notice of References Cited (PTO-892)

2) ☒ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) □ Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date _____.

4) □ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ .

5) □ Notice of Informal Patent Application

6) □ Other: _____.

DETAILED ACTION

1. Claims 1 – 48 are pending in the current application.

Allowable Subject Matter

2. Claims 9, 23 and 37 are objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims.

Response to Arguments

3. Applicant's arguments filed 09/11/2006 have been fully considered but they are not persuasive. In response to the Non-Final Office Action dated 06/15/2006, applicant argues:

(1) The claims require invoking an operating system command in the source node to transmit the message to the destination object within the source node if the destination object is in the source node. The cited col. 13 does not teach or suggest this claim requirement because the cited procedure 348 is used to send a message to a remote node, not to a destination object within the source node as claimed. [p. 15, lines 5 – 11];

(2) The claims also require invoking the operating system command in the destination node to transmit the message to the destination object within the destination node. The cited col. 13 does not teach or suggest this claim requirement because the cited procedure 348 is used to send a message to a remote node. [p. 15, lines 12 – 20];

(3) Talluri discusses an OpenBarrier and CloseBarrier procedures which differ from a remote read operation. The OpenBarrier procedure does not block continued operation of the executing thread whatsoever; and the CloseBarrier procedure blocks operation of the executing thread only until all pending remote write operations by the sending system's network interface (NIC card) have completed. [p. 15, lines 21 – 30];

(4) Talluri does not teach or suggest the claim requirements of an operating system command invoked on the source node or destination node depending on the determination of whether the source node and the destination node are the same node. [p. 16, lines 1 – 11];

(5) Talluri does not disclose a message queue associated with each object in each node. Instead, the cited col. 11 mentions a receive message queue for messages that are sent to Node B. [p. 17, lines 3 – 6];

(6) Talluri does not teach a message queue to store a message sent to a destination object in the same node as the source node. [p. 17, lines 11 – 17];

(7) Talluri does not teach the combination that the message packet includes the requirements of the intervening claims 3, 17, and 31 that the
message packet includes the determined address of the destination node and the address of the destination object. [p. 18, lines 1 – 7];

(8) Talluri does not teach the claim requirement of sending messages in the same destination node to a message queue associated with the destination object. [p. 18, lines 21 – 27];

(9) Talluri does not teach a destination network object that receives a message packet from another node invokes the operating system command to transmit the message to a message queue to the destination object within the destination node. [p. 19, lines 16 – 23];

(10) Talluri does not teach that the same identifier is used to identify the destination object when the destination and source nodes are the same as well as when they are different. [p. 20, lines 5 – 10]; and

(11) Nowhere does the cited Lea teach that the system comprises a storage library system and that a component node for an electro mechanical component comprises a component of a storage library system. [p. 21, lines 1 – 6].

As to argument (1), examiner respectfully disagrees and submits that Talluri discloses that the sending and receiving nodes may be located within the same server computer [terms "sender" (or sending node) and "receiver" (or receiving node) will both be used to refer to computers 210 within the server 202; col. 9, lines 18 – 29] or the sending and receiving node may be remote to each other [also be used to perform remote writes between any "true" client computer 204 and a server computer 210; col. 9, lines 18 – 29]. When the sending and receiving node are within the server, they are local to each other and the cited procedure 348 would send the message locally. In addition, Pettus also teaches transmitting the message to the destination object within the source node if the destination node is the source node [col. 5, lines 23 – 36]. Therefore, the combination of Pettus and Talluri teaches a operation system command to transmit the message to the destination object within the source node if the destination node is the source node.

As to argument (2), examiner disagrees and notes that Talluri teaches a message receive procedure 350 (which is preferably implemented as part of the operating system's kernel) for processing received messages (i.e., passing them to appropriate applications for processing)[col. 13, lines 5 – 45]. Therefore, the message receive procedure of Talluri meets the recited operating system command in the destination node to transmit the message to the destination object within the destination node.

As to argument (3), examiner disagrees. It appears that applicant is arguing that the combination of Pettus and Talluri does not teach a remote read operation. However, notes that the currently presented claims do not require a remote read operation. Insert stock paragraph.

In response to argument (4), examiner respectfully disagrees and notes that Talluri discloses that the sending and receiving nodes may be located within the same server computer [terms "sender" (or sending node) and "receiver" (or receiving node) will both be used to refer to computers 210 within the server 202; col. 9, lines 18 – 29] or the sending and receiving node may be remote to each other [also be used to perform remote writes between any "true" client computer 204 and a server computer 210; col. 9, lines 18 – 29]. When the sending and receiving node are within the server, they are local to each other and the cited procedure 348 would send the message locally. In addition, Pettus also teaches transmitting the message to the destination object within the source node if the destination node is the source node [col. 5, lines 23 – 36]. Talluri teaches a message receive procedure 350 (which is preferably implemented as part of the operating system's kernel) for processing received messages (i.e., passing them to appropriate applications for processing) [col. 13, lines 5 – 45]. Therefore, the message receive procedure of Talluri meets the recited operating system command in the destination node to transmit the message to the destination object within the destination node.

As to argument (5), examiner disagrees. It appears that applicant is suggesting that the claims require a dedicated queue for each object; however, the claims only recite "a message queue associated with each object". Even though each object has an associated queue, the objects can be associated with the same queue. Talluri teaches a received message queue for the node; therefore, each object in the receiving node is associated with a queue, the received message queue.

In response to argument (6), examiner disagrees and notes that Talluri teaches a message queue to store a message sent to a destination object in the same node as the source node [send buffer; col. 13, lines 6 – 45]. The send buffer in Talluri is a message queue in the source node that stores a message sent to a destination object.

In response to argument (7), examiner disagrees and submits that the remote function pointer in Pettus meet the recited address of the destination object and the resource identifier in Pettus meet recited node address because the resource identifier locates the network address [col. 10, lines 53 – 67] that corresponds to the desired network resource.

As to argument (8), examiner disagrees and notes that Talluri teaches a received message queue for the node and the destination object retrieves the message from the queue [col. 18, lines 55 – 65]. Therefore, Talluri teaches sending messages in the same destination node to a message queue associated with the destination object.

In response to argument (9), examiner disagrees and submits that Talluri teaches a destination network object that receives a message packet from another node invokes the operating system command [a reliable message sending procedure 348 (which is preferably implemented as part of the operating system 340) for sending messages to a remote node; col. 13, lines 5 – 45] to transmit the message to a message queue to the destination object within the destination node [col. 11, lines 49 – 67 of Talluri].

As to argument (10), examiner disagrees and submits that Pettus teaches using the same identifier to identify a destination object [function pointer; col. 13, lines 39 – 60 of Pettus].

In response to argument (11), examiner disagrees and notes that Lea teaches a storage library system and that a component node for an electro mechanical component comprises a component of a storage library system [col. 16, lines 20 – 25]. The UI library of Lea meets the recited storage library because the UI library provides for control of electro mechanical components [electronic devices; col. 11, lines 23 – 57 of Lea].

Claim Rejections - 35 USC § 103

4. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

5. This application currently names joint inventors. In considering patentability of the claims under 35 U.S.C. 103(a), the examiner presumes that the subject matter of the various claims was commonly owned at the time any inventions covered therein were made absent any evidence to the contrary. Applicant is advised of the obligation under 37 CFR 1.56 to point out the inventor and invention dates of each claim that was not commonly owned at the time a later invention was made in order for the examiner to consider the applicability of 35 U.S.C. 103(c) and potential 35 U.S.C. 102(e), (f) or (g) prior art under 35 U.S.C. 103(a).

6. Claims 1 – 7, 11 – 13, 15 – 21, 25 – 27, 29 – 35, 39 – 41 and 43 – 48 are rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Patent No. 6,223,217 to Pettus in view of U.S. Patent No. 6,615,383 to Talluri et al. [hereinafter Talluri], both references cited in the previous office action.

7. As to claim 1, Pettus teaches the invention substantially as claimed including allowing communication among processing nodes in a system [col. 5, lines 23 – 35], comprising:

receiving, in a source node [client node 610, Fig. 6; col. 9, line 57 – col. 10, line 16], a request from a source object executing in the source node ["caller" object which, once instantiated, accepts service requests from client objects; col. 5, lines 23 – 36 and col. 18, lines 4 – 28] to send a message to a destination object executing in a destination node [server node 650, col. 9, line 57 – col. 10, line 16], wherein each node includes a processor capable of multitasking multiple program objects [CPU 402 and 502, Figs. 4 and 5; col. 6, line 52 – col. 7, line 15] and a communication interface to transmit and receive data with the other nodes [network adapter 645 and 685, Fig. 6; col. 10, lines 16 – 33];

determining, in the source node, whether the destination node and source node are a same node [if the requests can be serviced locally or remotely; col. 5, lines 23 – 36 and col. 18, lines 50 – 63];

transmitting the message to the destination object within the source node if the destination node is the source node [If the requests can be serviced locally, then the caller object routes the request to a local service object; col. 5, lines 23 – 36]; and

if the destination node is not the source node [if the request is for a service which must be provided by a service object located on a remote server node, then the caller object provides high-level, "client-server communication" protocol requests to the remote node; col. 5, lines 30 – 36], performing:

(i) transmitting, with the source node, the message to the destination node through the communication interface [RPC objects also include a "dispatcher" object which is located at the remote service node and receives the incoming service requests; col. 5, lines 36 – 49 and col. 10, lines 16 – 25]; and

(ii) transmit the message to the destination object within the destination node [the dispatcher object 670 directs the request to an application program 652 for execution of the service; col. 10, lines 26 – 33].

Although Pettus teaches the invention substantially, Pettus does not specifically teach invoking an operating system command in the source node to transmit the message to the destination object within the source node if the destination node is the source node and invoking an operating system command in the destination node to transmit the message to the destination object within the destination node.

However, Talluri teaches invoking an operating system command in the source node to transmit the message to the destination object [a reliable message sending procedure 348 (which is preferably implemented as part of the operating system 340) for sending messages to a remote node] within the source node if the destination node is the source node [converting global addressed to local physical addresses and transmitting the data being written onto an internal bus of the receiving system using those local physical addresses, or storing the data in a FIFO; col. 11, lines 3 – 20] and invoking an operating system command in the destination node to transmit the message to the destination object within the destination node [a message receive procedure 350 (which is preferably implemented as part of the operating system's kernel) for processing received messages (i.e., passing them to appropriate applications for processing); col. 13, lines 5 – 45].

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to modify the invention of Pettus to incorporate the features of invoking an operating system command in the source node to transmit the message to the destination object within the source node if the destination node is the source node and invoking an operating system command in the destination node to transmit the message to the destination object within the destination node as taught by Talluri because this ensuring that each message is sent to the second computer once and only once while retaining a high level of message transmission reliability and using a "write only" message sending protocol to make such remote write operations efficient [col. 1, lines 5 – 12 of Talluri].

8. As to claim 2, Pettus as modified teaches invoking, in the source node, an operating system command to transmit the message to the destination object [application program 612 communicates with the operating system 620 at a higher level when requesting an RPC service; col. 10, lines 1 – 16 of Pettus], and associating a message queue with each object in each node [col. 11, lines 49 – 67 of Talluri], and wherein the invoked operating system command in the source node transmits the message to the message queue associated with the destination object [col. 13, line 63 – col. 14, line 42 of Talluri].

9. As to claim 3, Pettus teaches determining, in the source node, an address of the destination node that addresses the destination node when transmitting messages through the communication interface [network address; col. 10, lines 53 – 67];

generating, in the source node, at least one message packet [request data packet; col. 17, line 63 – col. 18, lines 5] including the message, the determined address, and an address of the destination object [function pointer; col. 13, lines 39 – 60]; and

transmitting, with the source node, the at least one message packet to the destination node over the communication interface [service request packet may be forwarded from the caller object; col. 17, line 64 – col. 18, line 5].

10. As to claim 4, Pettus as modified teaches the communication interface comprises a bus and wherein including the address of the destination node in the message causes the destination node to read the at least one message packet transmitted on the bus [col. 11, lines 20 – 35 of Talluri].

11. As to claim 5, Pettus as modified teaches determining, in the destination node, the destination object for the at least one message packet [child ID field which, in turn, indicates that the request is to be forwarded on to another dispatcher object; col. 17, line 64 – col. 18, line 5 of Pettus];

extracting, in the destination node, the message from the message packet [networking interface 680 reformats the request and forwards it to a dispatcher object 670; col. 10, lines 26 – 33 of Pettus], wherein the invoked operating system command in the destination node transmits the message to the message queue associated with the destination object [col. 13, line 63 – col. 14, line 42 of Talluri].

12. As to claim 6, Pettus as modified teaches invoking an operating system command [col. 10, lines 1 – 16 of Pettus], with the source object, to send the message to a message queue associated with a source network object in the source node [col. 14, lines 50 – 60 of Talluri];

determining, with the source network object, an address of the destination node that addresses the destination node when transmitting messages through the communication interface [network address; col. 10, lines 53 – 67 of Pettus];

generating, with the source network object, at least one message packet [request data packet; col. 17, line 63 – col. 18, lines 5 of Pettus] including the message, the determined address of the destination node, and an address of the destination object [function pointer; col. 13, lines 39 – 60 of Pettus];

transmitting, with the source network object, the at least one message packet to the destination node over the communication interface [service request packet may be forwarded from the caller object; col. 17, line 64 – col. 18, line 5 of Pettus]; and

receiving, with a destination network object, the at least one message packet [col. 17, line 64 – col. 18, line 5 of Pettus], wherein the destination network object invokes the operating system command in the destination node to transmit the message to a message queue associated with the destination object in the destination node [col. 13, line 63 – col. 14, line 42 of Talluri].

13. As to claim 7, this is rejected for the same reasons as claim 5 above.

14. As to claim 11, Pettus as modified teaches each object is assigned a unique object identifier in the system, and wherein the unique identifier is used within all nodes to identify the destination object to receive the message [col. 9, lines 40 – 55 of Talluri and col. 13, lines 39 – 60 of Pettus].

15. As to claim 12, Pettus teaches each node is assigned a unique node identifier [locates the network address that corresponds to the desired network resource; col. 10, lines 54 – 67] used within all nodes to identify the destination node to receive the message.

16. As to claim 13, Pettus teaches a function call ["caller" object which, once instantiated, accepts service requests from client objects; col. 5, lines 23 – 36 and col. 18, lines 4 – 28] receives the request from the source object to send the message to the destination object [server node 650, col. 9, line 57 – col. 10, line 16], determines whether the destination node is the same node [if the requests can be serviced locally or remotely; col. 5, lines 23 – 36 and col. 18, lines 50 – 63], sends the message to the destination object [If the requests can be serviced locally, then the caller object routes the request to a local service object; col. 5, lines 23 – 36] or causes the transmittal of the message to the destination node over the communication interface, and maintains the object and node identifier assignment [col. 10, lines 54 – 67], further comprising:

updating the node and object identifier used by each function call in each node to reflect a modification to the arrangement of nodes or objects in the system [col. 12, lines 28 – 40].

17. As to claim 43, Pettus as modified teaches the operating system command invoked to transmit the message to the destination object if the destination node is the source node and if the destination node is not the source node [col. 9, line 57 – col. 10, line 17 of Pettus] comprises a same operating system function [col. 13, lines 5 – 45 of Talluri].

18. As to claim 44, Pettus as modified teaches the operating system command invoked in the source node in response to determining that the destination node and the source node are the same queues the message in a message queue of the destination object [col. 11, lines 49 – 67 of Talluri], further comprising:

invoking the operating system command in the source node to queue the message in a communication interface object queue in response to determining that the destination node is not the source node [col. 16, lines 20 – 32 of Talluri], wherein a communication interface object transmits the message from the communication interface object queue to the destination node [col. 13, line 63 – col. 14, line 42 of Talluri].

19. As to claims 15 – 21, 25 – 27 and 45 – 46, these are system claims that correspond to method claims 1 – 7, 11 – 13 and 43 – 44; note the rejections to claims 1 – 7, 11 – 13 and 43 - 44 above, which also meet these system claims.

20. As to claims 29 – 35, 39 – 41 and 47 – 48, these are rejected for the same reasons as claims 1 – 7, 11 – 13 and 43 – 44 above.

21. Claims 8, 10, 14, 22, 24, 28, 36, 38 and 42 are rejected under 35 U.S.C. 103(a) as being unpatentable over Pettus and Talluri further in view of U.S. Patent No. 6,349,352 to Lea [cited in the previous office action].

22. As to claim 8, Pettus as modified does not specifically teach a component node that controls an electro-mechanical component of the system and manages system commands and the message includes a command to instruct a motion object in the component node to control the electromechanical component to perform an operation.

However, Lea teaches a first node comprises a controller node [Intermediate AV nodes; col. 8, lines 17 – 32 of Lea] and at least one second node [Full AV nodes; col. 8, lines 6 – 17 of Lea] comprises a component node that controls an electro-mechanical component [electronic device; col. 11, lines 23 – 57 of Lea] of the system, wherein the source object comprises a work management object [Device manager 761; col. 15, lines 41 – 45 of Lea] in the controller node that manages system commands [Device manager 761 is responsible for creating and managing the DCMs that represent devices managed by an FAV device; col. 15, lines 40 – 45 of Lea] and the message includes a command [col. 16, lines 17 – 21 of Lea] to instruct a motion object [Device Modules 720; col. 15, lines 44 – 50 of Lea] in the component node to control the electromechanical component to perform an operation [each DCM functions as a control point for a device; col. 15, lines 44 – 50 of Lea].

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to apply the teaching of a component node that controls an electro-mechanical component of the system and manages system commands and the message includes a command to instruct a motion object in the component node to control the electromechanical component to perform an operation as taught by Lea to the invention of Pettus as modified because this allows new devices to be integrated into an existing network and provide their services in a seamless manner [col. 5, lines 50 – 55 of Lea].

23. As to claim 10, Pettus as modified teaches the system comprises a storage library system, and the electromechanical component comprises a component of a storage library system [col. 16, lines 20 – 25 of Lea].

24. As to claim 14, Pettus as modified teaches each node transmits signals to determine an availability of other nodes on the communication interface [col. 27, lines 16 – 24 of Lea].

25. As to claims 22, 24 and 28, these are system claims that correspond to method claims 8, 10 and 14; note the rejections to claims 8, 10 and 14 above, which also meet these system claims.

26. As to claims 36, 38 and 42, these are rejected for the same reasons as claims 8, 10 and 14 above.

Conclusion

27. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

CONTACT INFORMATION

28. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Li B. Zhen whose telephone number is (571) 272-3768. The examiner can normally be reached on Mon - Fri, 8:30am - 5pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, William Thomson can be reached on 571-272-3718. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Li B. Zhen
Examiner
Art Unit 2194

LBZ

MENG-AL T. AN
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100